Inter-Domain Path Authentication in Tactical MANETs

Steffen Reidt† and Mudhakar Srivatsa‡
Royal Holloway, University of London†
IBM T.J. Watson Research Center‡
s.reidt@rhul.ac.uk, msrivats@us.ibm.com

***Abstract*—This paper presents a lightweight probabilistic path authentication scheme for mobile ad hoc networks (MANETs) based upon a new cryptographic primitive *composite MAC*. The proposed path-authentication scheme allows us to reliably identify nodes on a route over which a sequence of packets traverses. This path-authentication scheme is robust against selfish or malicious nodes that do not follow the scheme. Furthermore, it allows us to detect, and up to a certain accuracy pinpoint, any misbehaving node that deviates from the correct forwarding behavior. In our scheme, composite MAC can have any length starting from one bit. This flexibility allows the proposed scheme to strike various trade-offs depending on the constraints imposed by the MANET and the desired security properties. We provide an informal security analysis and argue that a short MAC can be sufficient to authenticate paths with high probability.**

I. Introduction

Mobile ad hoc networks (MANETs) have been developed to support communication in tactical and other situations where the availability of a fixed communication infrastructure cannot be assumed. Many such situations require resources of a coalition wherein multiple groups and organizations come together, communicate, and collaborate, all within a short period of time; for example, in a disaster recovery operation, the local police force, fire-fighters, military forces, medical crews, and other organizations may all coordinate their activities. Such situations call a *coalition MANET*, an interconnect of several MANETs governed by different administrative domains, to enable the end-to-end communication. This, in turn, requires *inter*-domain routing, referred to as IDRM (Inter-Domain Routing for MANETs) in shorthand[3], that are now being actively researched.

Inter-domain routing opens up numerous security challenges that arise from interactions between multiple management domains. There are three general classes of security threats for IDRM: attacks on the protocol itself, falsification of the information exchanged in the protocol (falsification attack), and forwarding traffic along a different path than the one identified by the routing protocol (incorrect forwarding). Attacks against the protocol itself include attempts to spoof the network identity of IDRM routers, compromise the integrity of routing protocol messages exchanged between IDRM routers, etc. These attacks are the simplest to address since it is essentially a matter of establishing a secure channel between two cooperating entities (neighboring IDRM routers). Falsification attacks attempt to inject false information in the routing protocol and thus introduce routing anomalies such as black holes, grey holes, sub-optimal routes, etc. This problem has been well explored in the context of inter-domain routing protocols for the Internet, and several schemes, such as Secure BGP (S-BGP [13]), Secure Origin BGP (so-BGP [27]), and Pretty Secure BGP (ps-BGP [26]), have been proposed to address the problem.

In this paper, we focus on the third class of attacks: incorrect forwarding. A malicious node on the route can forward packets incorrectly to interrupt critical data flows or divert traffic to perform timing and traffic analysis attacks. In fact, many falsification attacks also result in incorrect forwarding, making it an important behavior to detect. Another reason for detecting incorrect forwarding is misconfigured or selfish nodes. To elaborate, policy-based routing plays a critical role in IDRM [3] in ensuring overall end-to-end network performance, reliability, and security. A misconfigured or buggy node may forward packets incorrectly resulting in degradation of these end-to-end network qualities. For these reasons, the focus of this paper is on detecting and diagnosing incorrect forwarding behavior in a coalition MANET.

Recently, Boldyreva *et al.*[2] developed cryptographic signature schemes that can be used to monitor node forwarding behavior in inter-domain routing protocols. However, their scheme, designed for an Internet-like setting, incurs substantial communication and computa-

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
01 DEC 2008	**N/A**	**-**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Inter-Domain Path Authentication in Tactical MANETs	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Royal Holloway, University of London**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release, distribution unlimited

13. SUPPLEMENTARY NOTES
See also ADM002187. Proceedings of the Army Science Conference (26th) Held in Orlando, Florida on 1-4 December 2008, The original document contains color images.

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**	**UU**	**8**	

tion overhead making it unsuitable for a MANET. In particular, the scheme requires each node on a path to sign the message of each forwarded packet incurring a substantial computation overhead, and the length of the signature is at least 160 bits incurring a substantial communication overhead. A probabilistic path-authentication scheme that requires each node to sign only a fraction of the forwarded packets can reduce this overhead; however, such schemes need to use a signature with a designated verifier, i.e., only the destination node is able to verify the signature. This property is crucial, since a malicious node on the route could otherwise detect the packets that were signed, and selectively modify or drop them. Traditionally, signature schemes with a designated verifier are based on public key cryptography, however an interesting alternative is to use message authentication codes (MACs). MACs, such as HMAC by Krawczyk, Bellare, and Canetti [15], are calculated by an algorithm that involves evaluating a cryptographic hash function in combination with a secret key. MACs are therefore computationally efficient. Furthermore, they can have any desired length starting at one bit making them bandwidth efficient.

In this paper, we introduce a new cryptographic primitive, *composite MAC*, which forms the basis of our lightweight probabilistic path-authentication scheme. Composite MAC is an extension of Katz's and Lindell's [12] aggregate MAC. As the name implies, a composite MAC is a composition of MACs which rely on the existence of secret keys. Therefore, for our path-authentication scheme, we require that each node on a route shares a symmetric key with the destination node. If every node in the network is a potential destination node, then each pair of nodes has to share a symmetric key. While this appears to be a strong assumption, we note that the number of organizations that participate in a coalition MANET is small (< 20), and symmetric keys can be shared only between the organizations. The overhead of setting these symmetric keys is less of an issue for an "organized" MANET where offline pre-configuration during a mission planning phase is expected. We further explore a back tracing technique for the proposed composite MAC scheme that can identify forwarding nodes even if they differ from the expected ones. This property is of particular interest in MANETs where routes are likely to change due to mobility and wireless communication.

The rest of this paper is organized as follows. Section II compares coalition MANETs, intra-organizational MANETs, and inter-domain routing on the Internet, highlighting their differences. Section III describes our probabilistic path authentication scheme which is followed by an informal security analysis of the scheme in Section IV. Finally, Section V concludes the paper.

II. Related Work

A. Intra-domain Routing Security in MANETs

Several authors have described solutions that attempt to mitigate falsification attacks in intra-domain routing protocols for MANETs. Intra-domain routing protocols can be broadly classified into *proactive* and *reactive* routing protocols. SAODV [7][6] provides an example of reactive routing protocol security. It uses hash chains to avoid manipulation of hop counts in route discovery messages, and digital signatures are used for the immutable parts of these messages, to provide end-to-end confirmation that the request reached the owner of the address. SLSP [19] is an example of a security mechanism for a proactive routing protocol. It uses signatures on link state update messages to avoid manipulation of the topology information. The SAODV solution is focused on verifying the validity of the path, whereas the SLSP approach is based around determining the correctness of the network topology. In both cases, the existence of a Public Key Infrastructure (PKI) is assumed. Other research has explored the possibility of using Identity-based Public Key Cryptography (ID-PKC) [14].

Recently, several research proposals have used cooperative network monitoring based on root cause analysis techniques to detect malicious and faulty nodes in networks. Cooperative monitoring techniques range from physical layer power estimation for detecting jamming attacks [28][9], MAC layer misbehavior detection [20][16] to routing layer faults and anomaly detection [25]. However, to date, all cooperative root cause analysis techniques assume that the monitors are honest. While this is a reasonable assumption for an intra-domain setting wherein all monitors belong to a single domain, an inter-domain setting is faced with the challenge of handling faulty monitors that may be malicious, rational-selfish or Byzantine.

B. Inter-domain Routing Security in the Internet

Inter-domain routing in the Internet is managed using BGP4 (Border Gateway Protocol) [22]. This was originally developed for use in a trusted environment, and so provides little security against attackers or misconfiguration. The need for additional security mechanisms has been recognized in recent times, and demonstrated by the AS7007 incident [17][18] and more recent "hijacking" of

a part of the YouTube address space [23]. Both incidents are believed to have been due to misconfiguration, rather than malicious intent. Current BGP operations depend completely on peers trusting one another not to inject bad information into the routing updates. This is coupled with limited filtering (e.g. to filter out advertisements of unallocated address space, and to ensure that downstream customers only advertise their own address prefixes). In addition to such filtering, there is some use of TCP-MD5 [8] to provide integrity protection for the protocol between peer routers.

There have been a number of different proposals for adding security to BGP, such as S-BGP [13], Secure Origin BGP (so-BGP) [27], and Pretty Secure BGP (ps-BGP) [26]. These competing proposals, embody different views on the appropriate model for authenticating ownership of identifiers (such as AS numbers and prefixes). These solutions tend to rely heavily on public key signatures, although some attempts are made to ensure that results of signature verification can be cached. Both the computational burden and the key and certificate storage requirements are significant for a protocol operating on an Internet scale. To address this, other proposals have been made where such signature use is minimized, e.g., secure path vector (SPV) [10][21].

Approaches to BGP security which avoid the use of cryptographic components by relying on BGP policy tools have also been proposed. One solution, pgBGP (Pretty Good BGP) [11], simply adjusts BGP policies to provide some additional cautiousness in accepting new routes. New origin ASs for a prefix are regarded as suspicious for a period of time, and then accepted as normal. This reduces the likelihood of a (short-lived) prefix or sub-prefix hijacking being successful when used in conjunction with appropriate monitoring systems. RPSL (Route Policy Specification Language) [1] provides a way for ISPs to describe their routing policies. For example, it will indicate what routes they accept from a particular neighboring AS, and what routes they advertise to them. This information is stored in one of a number of central databases, and can be automatically extracted to perform path selection on a router. However, deployment is limited and in practice this information tends to be stale and at best provides some hints on the selected path.

C. Monitoring Forwarding Behavior

Boldyreva *et al.* [2] introduced the new primitive of an *ordered multi-signature* (OMS) scheme, which allows signers to attest to a common message as well as the order in which they signed it. The benefit of Boldyreva's scheme compared to previous similar work on multi-signatures (MS) is that it does not require synchronized clocks or a trusted first signer. They focus on path authentication in the Internet as the main application of their scheme. Pairing based signature schemes (as Boldyreva's) have a signature size of typically 60 bytes, which is still small compared to other public key based signature schemes. Since the typical packet size is 1500 bytes, in wired as well as in wireless communication, the additional communication overhead caused by the 60 byte signature is approximately 5% (for 1200 byte payload). We note that most nodes in a MANET are battery powered and thus severely constrained. Hence, while this additional communication overhead might be feasible for the Internet, decreasing the lifetime of a MANET by 5% appears to be unreasonable. Furthermore, performing elliptic curve operations on each forwarding node for each packet imposes a computational overhead, which is infeasible for devices with limited computational capabilities and battery power.

In this paper, we propose a light weight probabilistic path authentication scheme using aggregate MACs as introduced by Katz and Lindell [12] (summarized in Section III). Our scheme incurs low communication overhead (4-8 bits per packet), low computation costs (MAC computation) and is highly responsive (a short stream of 20 packets is sufficient to authenticate a path of length 5 with high probability).

III. Probabilistic Path Authentication Scheme

In this section we introduce our probabilistic MAC path authentication scheme, which uses *composite MACs*, an extension of aggregate MACs introduced by Katz and Lindell [12] for message authentication. We exploit the nice properties of aggregation, while shortening the MAC size to a small number of say 4 to 8 bits. We note that shortening the MAC size and thus the length of the authentication tag yields only probabilistic results. For example, a verified tag of length 4 can only ensure authenticity with a probability of $\frac{15}{16}$. The scheme will however extract its strength by aggregating the information contained in multiple authentication tags that are embedded in multiple packets. The analysis of packets for path authentication is performed on a per packet basis. The proposed scheme is agnostic to packet losses and out of order packet arrivals; only the total number of packets used for the authentication is of interest. Hence, composite MACs are especially useful

in a MANET setting where communication is unreliable and highly expensive.

A basic requirement for the usage of MACs is the existence of symmetric keys. For our scheme we require, that each node on a route shares a symmetric key with the destination node. If every node in the network is a potential destination node, then consequently each pair of nodes has to share a symmetric key. As discussed in the introduction, this is a reasonable assumption in "organized" MANETs, where such keys can be distributed off-line once during the mission planing phase. Key distribution schemes that require minimal storage and only constant communication overhead include non-interactive key distribution schemes as proposed by Sakai *et al.*[24], or for a hierarchically organized network by Gennaro *et al.*[4]. We propose the usage of such a non-interactive key distribution scheme, where a central authority needs to distribute only one secret key to each node in the network to equip each pair of nodes with a shared key.

In this section, we will first recall Katz's and Lindell's aggregate MAC, and show how it can be easily extended to an Ordered aggregate MAC. We then define our composite MAC as an extension of the aggregate MAC scheme, which especially allows the detection of Byzantine adversaries. Robustness against a Byzantine adversary is vital, since an adversary could otherwise easily subvert the aggregate MAC scheme by overwriting the tag with random content. Since the remaining nodes on the route would aggregate their MACs with a random tag, the resulting tag would still remain random, and therefore be of no use for the destination node. While we cannot stop an adversary from overwriting the tag, we beat him at his own game, and incorporate overwriting of the tag in the composite MAC. Honest nodes who are positioned between the Byzantine node and the destination node in the route, and overwrite the tag with their MAC as part of the protocol, allow us to detect the Byzantine nodes with non-trivial probability.

A. Composite MACs

We first recall Katz's and Lindell's construction for an aggregate MAC. While in Katz's and Lindell's scheme the message m_i can be different for each node i, the message m in our scheme is the same for all nodes. Our definition of an aggregate MAC 1 is therefore Katz's and Lindell's definition for the construction of an aggregate MAC with $m_i = m, \forall i$. We then show how an aggregate MAC can be easily extended to an ordered aggregate MAC and a composite MAC. We use $k_{i,d}$ to denote the shared key between node i and node d.

Definition 1 (Aggregate MAC): Let Mac be a pseudorandom MAC, that takes a key $k_{i,d}$ and the actual message m as input. tag is the authentication tag of the same length as Mac.

- **Initialisation:** The sender sets
$$\mathsf{tag} = \mathsf{Mac}_{k_{s,d}}(m)$$
where $k_{s,d}$ is the shared key between the sender s and the destination node d. The sender forwards tag and the message m.
- **Aggregation:** On input m and tag, a node i sharing the key $k_{i,d}$ with the destination node, computes
$$\mathsf{tag} = \mathsf{tag} \oplus \mathsf{Mac}_{k_{i,d}}(m)$$
Node i forwards tag and the message m.
- **Verification:** On input m, tag and an expected set I of nodes that aggregated their MAC to tag (including the sender), the destination node d verifies:
$$\mathsf{tag} = \bigoplus_{i \in I} \mathsf{Mac}_{k_{i,d}}(m)$$

The aggregate MAC can easily be modified to an Ordered Aggregate MAC:

Definition 2 (Ordered Aggregate MAC): Let Mac be a pseudorandom MAC, that takes a key $k_{i,d}$ and the actual message m as input. tag is the authentication tag of the same length as Mac.

- **Initialisation:** The sender sets
$$\mathsf{tag} = \mathsf{Mac}_{k_{s,d}}(m)$$
where $k_{s,d}$ is the shared key between the sender s and the destination node d. The sender forwards tag and the message m.
- **Aggregation:** On input m and tag, a node i sharing the key $k_{i,d}$ with the destination node, computes
$$\mathsf{tag} = \mathsf{Mac}_{k_{i,d}}(m, tag)$$
Node i forwards tag and the message m.
- **Verification:** On input m, tag and an expected ordered set $I = \{s, i_1, i_2, \ldots, i_k\}$ of nodes that aggregated their MAC to tag, the destination node d verifies:
$$\begin{aligned}\mathsf{tag} \;=\; & \mathsf{Mac}_{k_{i_k,d}}(m, \mathsf{Mac}_{k_{i_{k-1},d}}(m, \ldots, \\ & \mathsf{Mac}_{k_{i_1,d}}(m, \mathsf{Mac}_{k_{s,d}}(m))))\end{aligned}$$

Both aggregate MACs as defined in Definitions 1 and 2 are vulnerable against a Byzantine adversary (as

described earlier in Section III). While we cannot stop an adversary from overwriting the tag, we extend the aggregate MAC (from Definition 1) to incorporate overwriting of the authentication tag in the composite MAC scheme. The key intuition is that even if a Byzantine node i_j in a route $\{s, i_1, i_2, \ldots, i_r\}$ ($j < r$) replaces the tag with random content, overwritings by subsequent nodes $\{i_{j+1}, \ldots, i_r\}$ allow the recipient to detect (and identify) the Byzantine node i_j.

Definition 3 (Composite MAC): Let Mac be a pseudorandom MAC, that takes a key $k_{i,d}$ and the actual message m as input. tag is the authentication tag of the same length as Mac.

- **Initialisation:** The sender sets

  $$\mathsf{tag} = \mathsf{Mac}_{k_{s,d}}(m)$$

  where $k_{s,d}$ is the shared key between the sender s and the destination node d. The sender forwards tag and the message m.
- **Composition:** On input m and tag, a node i sharing the key $k_{i,d}$ with the destination node, computes

  $$\mathsf{tag} = \mathsf{tag} \circ \mathsf{Mac}_{k_{i,d}}(m)$$

  Node i forwards tag and the message m. The composition operator $\circ$ can be defined as $Aggregate$, $Overwrite$, or $KeepIdentical$:
  Aggregate: $\mathsf{tag} \circ \mathsf{Mac}_{k_{i,d}}(m) = \mathsf{tag} \oplus \mathsf{Mac}_{k_{i,d}}(m)$
  Overwrite: $\mathsf{tag} \circ \mathsf{Mac}_{k_{i,d}}(m) = \mathsf{Mac}_{k_{i,d}}(m)$
  KeepIdentical: $\mathsf{tag} \circ \mathsf{Mac}_{k_{i,d}}(m) = \mathsf{tag}$
- **Verification:** On input m, tag and an expected ordered set I of nodes that modified the tag (including the sender), the destination node d verifies:

  $$\mathsf{tag} = \bigcirc_{i \in I} \mathsf{Mac}_{k_{i,d}}(m)$$

A composite MAC as defined in Definition 3 is agnostic to selfish nodes on the route. We say that a node is selfish if it simply ignores the path authentication scheme, i.e. leaves the tag unchanged to save energy for example. Since selfish nodes put no information at all in the authentication tag, evidence about their existence in the route has to be provided by other nodes. The only reliable information that a node has about other nodes on the path, is the identity of the prior node that forwarded the packet to it. Routing tables, giving information about other nodes on the route, do not necessarily reflect the real packet forwarding route. Also, in a wireless broadcast medium, the subsequent (next hop) node on the route might not be the intended one. For example, a node A might forward a packet to an intended next hop node B; however, a node C might intercept the packet and interpose itself on the path from A to B (or even bypass node B). In order to detect selfish nodes, we therefore incorporate the information about the respective prior node $i-1$ as an additional parameter in the MAC. We use F to denote a pseudorandom function that takes the message m and the key $k_{i,d}$ as input, and outputs a unique string of the same length as tag for each key $k_{i,d}$ and message m:

$$\mathsf{Mac}_{k_{i,d}}(m, ID_{i-1}) = F(m, k_{i,d}, ID_{i-1}) \qquad (1)$$

Thus, if a node that was expected to be part of a route did not aggregate its MAC to an authentication tag when it was expected to, the destination node can identify the missing node with a non-trivial probability.

In the following sections of this paper, we use composite MAC to denote a composite MAC from definition 3 based on a MAC as defined in equation 1.

B. Back Tracing

In this section, we describe our back tracing technique and present two enhancements to the composite MAC scheme to facilitate efficient back tracing. Let $\mathcal{S}$ denote the set of nodes that may potentially modify the authentication tag (in the worst case, $\mathcal{S}$ is the set of all nodes in the coalition network). Back tracing is achieved by computing the authentication tag for all combinations of $2^{|\mathcal{S}|}$ possible MACs. Hence, the worst case complexity of back tracing is $O(2^{|\mathcal{S}|})$. In practice, we limit back tracing up to a limited depth $d \ll |\mathcal{S}|$, thereby considerably reducing the complexity of back tracing at the cost of decreasing the efficacy of back tracing.

Given that the worst case complexity of back tracing may be exponential we apply two enhancements to facilitate efficient back tracing. Below, we describe these enhancements. First, we split an authentication tag in sub-tags. If a tag is divided in sub-tags, then each sub-tag is handled separately as if it was a normal tag of full length. To this end, the respective MAC $\mathsf{Mac}_{i,d}$ of length n is divided in c_n MACs $\mathsf{Mac}_{i,d,j}, j = 1 \ldots c_n$ of length n/c_n such that: $\mathsf{Mac}_{i,d} = \mathsf{Mac}_{i,d,1}|\mathsf{Mac}_{i,d,2}| \ldots |\mathsf{Mac}_{i,d,c_n}$ ($|$ is the concatenation operator). Splitting the tag in sub-tags facilitates to incorporate evidence about all nodes on a path in fewer packets. If a tag is split in 4 sub-tags for example, then the total number of tags available for the analysis increases by a factor of 4. The drawback of shorter tags however, is a smaller probability for a unique and back-traceable tag. We therefore leave the the number of sub-tags that each tag is devided in as a

parameter that can be configured to suit the respective requirements for the path authentication scheme.

Second, we pseudo-randomly choose only a small subset of nodes on the route to aggregate or overwrite the authentication tag on a per-packet basis. We ensure that the choice of a node to aggregate, overwrite or keep an authentication tag identical, is known by the respective forwarding node and the destination node, and must not be known by any other node in the network. Assuming that a large majority of nodes are good, this approach significantly decreases the number of possible nodes that modify the authentication tag, thereby significantly decreasing the cost of back tracing. At the same time, it is not possible for a bad node to selectively misbehave (and avoid detection) since it cannot a priori guess the choice of composition (aggregate / overwrite / keep identical) exercised by the good nodes on the forwarding route.

We use parameters p and q to denote the fraction of sub-tags that are modified by aggregation and overwriting, respectively. Consequently, $1-p-q$ denotes the fraction of sub-tags that is kept identical by the node. To achieve these properties, we let a node i aggregate its MAC to the j'th sub-tag of a packet if:

$$PRF(pID, k_{i,d}, j) \leq p \cdot 10^{\lambda} \mod 10^{\lambda}$$

overwrite the tag with its MAC if:

$$p \cdot 10^{\lambda} < PRF(pID, k_{i,d}, j) \leq (p+q) \cdot 10^{\lambda} \mod 10^{\lambda}$$

and keep it identical otherwise, where PRF is a publicly known pseudorandom function, and $k_{i,d}$ is the shared key between node i and the destination node. The exponent λ controls the possible accuracy of p, i.e. $p \in [0,1] \subset \mathbb{R}$ can be expressed with an accuracy of λ decimal places. The packet identifier pID can be any part of the packet that uniquely defines the packet. Depending on the routing protocol this could be a sequence number, or the timestamp on the packet. Using pID essentially allows us to pseudo-randomly change the choice of composition on a per-packet basis.

IV. Security Properties of Composite MAC

A. Unforgeability and Randomness

Katz and Lindell have proven that aggregate MACs are unforgeable under an adaptive chosen-message attack [5]. The attacker in their security model is allowed to have all but one of the shared keys between the nodes aggregating a message and the destination node. The only requirement is, that the individual MACs are unpredictable. This however holds for any secure (standard) MAC, by the definition of security for MACs.

A composite MAC that is overwritten by one or several MACs is the same as an aggregate MAC that has the last overwritten MAC as its initial value. Nodes keeping the composite MAC identical do not change anything and can be ignored for the security analysis. Therefore, composite MACs are just aggregate MACs with a possibly altered start value. Since the start value of an aggregate MAC can be any MAC, this does not affect the security of composite MACs. Consequently, unforgeability under an adaptive chosen-message attack follows directly from Katz's and Lindell's proof for aggregate MACs. The attacker model however cannot allow the attacker to have all but one of the shared keys anymore. The restriction that needs to be made is that the attacker does not have one of the keys that belongs to the last overwriting or one of the nodes after the last overwriting node that aggregate their MAC, In order to forge the composite MACs authenticating a complete path however, the attacker needs to have the keys of all nodes on the path.

Besides the unforgeability, a composite MAC used for path authentication needs to be pseudorandom. As described in Section III-A, nodes leave a certain ratio of composite MACs unchanged. If an attacker knew whether the former nodes on the route modified the composite MAC, it could selectively drop packets or overwrite the composite MAC. Dropping the packets with modified composite MACs could totally bypass path authentication, and selective overwriting of composite MACs could be used to accuse honest nodes on the path. These kinds of attacks are not possible for a pseudorandom composite MAC. An attacker can still drop or overwrite the MAC, but not selectively; this consequently reveals his bad behavior with non-trivial probability.

B. Detection of Selfish and Byzantine Adversaries

Unforgeability and randomness of composite MACs ensure that no node except of the destination node can learn any information from a received tag or create a valid tag on behalf of other nodes. While these are necessary security properties, hostile nodes have several other possibilities to disable the authentication tag or to bypass it. Since nodes are not able to forge the authentication tag in a meaningful way, the only things they can do is to: (a) follow the protocol correctly, (b)

leave the tag unchanged, and (c) change the tag in a way that makes it unreadable for the destination node.

Due to the randomness of the composite MAC, the strategies (a), (b) and (c) cannot be selectively applied on packets. Thus, if a node is switching between these strategies, it can have no better tactic than switching randomly between packets. The analysis of the tags, i.e. verification or detection of malicious behavior, is performed on a per packet basis (equivalent per authentication tag basis). Thus, analyzing several tags will result in a stack of information about each node. If a node switches between strategies (a), (b) and (c), this will be reflected in inconsistent evidences about this node. The proposed scheme therefore tolerates nodes which are switching their strategies, the results will simply apply in the ratio they run the respective strategy.

V. Conclusion

In this paper, we have proposed a novel probabilistic path-authentication scheme for detecting (and diagnosing) incorrect forwarding behavior in a coalition MANET. Our scheme is highly efficient — it requires only a small ratio of the forwarding nodes to sign the packet, and it can work with an authentication tag (signature) of length four or eight bits. We have developed techniques that allow the designated recipient to detect, with high probability, incorrect forwarding behavior by aggregating these short signatures. The recipient can backtrace on an authentication tag, to reveal the signers' identities and identify misbehaving nodes with non-trivial probability. We have presented an informal security analysis of our proposed scheme and argued that using small MACs can be sufficient to authenticate paths with high probability.

Acknowledgements

The authors would like to acknowledge contributions of Dakshi Agrawal in formulating the problem and guiding the search for a solution. Kang-won Lee and H. Starsky Wong also played a crucial part in defining the problem area. The authors are grateful to Shane Balfe, Kenny Paterson, and Stephen Wolthusen for their guidance, mentoring, and support. Our work greatly benefited from many discussions with our colleagues, and without the benefit of their collaboration, this work would not have achieved its current form.

Research was sponsored by the U.S. Army Research Laboratory and the U.K. Ministry of Defence and was accomplished under Agreement Number W911NF-06-3-0001. The views and conclusions contained in this document are those of the author(s) and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Army Research Laboratory, the U.S. Government, the U.K. Ministry of Defence or the U.K. Government. The U.S. and U.K. Governments are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

References

[1] C. Alaettinoglu, C. Villamizar, E. Gerich, D. Kessens, D. Meyer, T. Bates, D. Karrenberg, and M. Terpstra. Routing Policy Specification Language (RPSL). RFC 2622 (Proposed Standard), June 1999. Updated by RFC 4012.

[2] Alexandra Boldyreva, Craig Gentry, Adam O'Neill, and Dae Hyun Yum. Ordered multisignatures and identity-based sequential aggregate signatures, with applications to secure routing. In *CCS '07: Proceedings of the 14th ACM conference on Computer and communications security*, pages 276–285, New York, NY, USA, 2007. ACM.

[3] Chi-Kin Chau, Jon Crowcroft, Kang-Won Lee, and Starsky H.Y. Wong. IDRM: Inter-Domain Routing Protocol for Mobile Ad Hoc Networks. Technical Report UCAM-CL-TR-708, University of Cambridge, Computer Laboratory, January 2008.

[4] Rosario Gennaro, Shai Halevi, Hugo Krawczyk, Tal Rabin, Steffen Reidt, and Stephen D. Wolthusen. Strongly-resilient and non-interactive hierarchical key-agreement in manets. Cryptology ePrint Archive, Report 2008/308, 2008. http://eprint.iacr.org/.

[5] Sha Goldwasser, Silvio Micali, and Ronald L. Rivest. A digital signature scheme secure against adaptive chosen-message attacks. *SIAM Journal on Computing*, 17:281–308, 1988.

[6] Manel Guerrero Zapata. Secure ad hoc on-demand distance vector (saodv) routing, September 2006. Internet Draft draft-guerrero-manet-saodv-06.txt.

[7] Manel Guerrero Zapata and N. Asokan. Securing Ad hoc Routing Protocols. In *Proceedings of the 2002 ACM Workshop on Wireless Security (WiSe 2002)*, pages 1–10, September 2002.

[8] A. Heffernan. Protection of BGP Sessions via the TCP MD5 Signature Option. RFC 2385 (Proposed Standard), August 1998.

[9] I. W. Ho, B. Ko, M. Zafer, C. Bisdikian, and K. Leung. Cooperative Transmit-Power Estimation in MANETs. In *Processings of IEEE WCNC 2008*, Las Vegas, March 2008.

[10] Yih-Chun Hu, Adrian Perrig, and Marvin Sirbu. SPV: secure path vector routing for securing BGP. *ACM SIGCOMM Computer Communications Review*, 34(4), October 2004.

[11] Josh Karlin, Stephanie Forrest, and Jennifer Rexford. Pretty Good BGP: Improving BGP by Cautiously Adopting Routes. In *The 14th IEEE International Conference on Network Protocols*, November 2006.

[12] Jonathan Katz and Andrew Y. Lindell. Aggregate Message Authentication Codes. In *Topics in Cryptology - CT-RSA 2008*, 2008.

[13] Stephen Kent, Charles Lynn, and Karen Seo. Secure Border Gateway Protocol (Secure-BGP). *IEEE Journal on Selected Areas in Communications*, 18(4):582–592, April 2000.

[14] A. Khalili, J. Katz, and W.A. Arbaugh. Toward secure key distribution in truly ad-hoc networks. In *Proceedings of 2003 Symposium on Applications and the Internet Workshops*, pages 342–346, January 2003.

[15] H. Krawczyk, M. Bellare, and R. Canetti. Hmac:keyed-hashing for message authentication, rfc 2104, Feb 1997.

[16] Y. W. Law, L. van Hoesel, J. Doumen, P. Hartel, and P. Havinga. Energy-efficient link-layer jamming attacks against wireless sensor network mac protocols. In *SASN '05: Proceedings of the 3rd ACM workshop on Security of ad hoc and sensor networks*, pages 76–88, New York, NY, USA, 2005. ACM.

[17] Ratul Mahajan, David Wetherall, and Tom Anderson. Understanding BGP misconfiguration. In *SIGCOMM '02: Proceedings of the 2002 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 3–16, New York, NY, USA, 2002. ACM.

[18] S. A. Misel. Wow, AS7007! http://www.merit.edu/mail.archives/nanog/1997-04/msg00340.html, April 1997.

[19] P. Papadimitratos and Z.J. Haas. Secure link state routing for mobile ad hoc networks. In *Proceedings of IEEE Workshop on Security and Assurance in Ad Hoc Networks, in 2003 International Symposium on Applications and the Internet Workshops, 2003*, pages 379–383, January 2003.

[20] S. Radosavac, J. H. Baras, and I. Koutsopoulos. A framework for MAC misbehavior detection in wireless networks. In *WiSE*, 2005.

[21] Barath Raghavn, Saurabh Panjwani, and Anton Mityagin. Analysis of the SPV Secure Routing Protocol: Weaknesses and Lessons. *ACM SIGCOMM Computer Communication Review*, 37(2), April 2007.

[22] Y. Rekhter, T. Li, and S. Hares. A Border Gateway Protocol 4 (BGP-4). RFC 4271 (Draft Standard), January 2006.

[23] RIPE NCC. YouTube Hijacking: A RIPE NCC RIS case study. http://www.ripe.net/news/study-youtube-hijacking.html, February 2008.

[24] R. Sakai, K. Ohgishi, and M. Kasahara. Cryptosystems based on pairing. In *sCIS 2000: The 2000 Symposium on Cryptography and Information Security*, 2000.

[25] M. Srivatsa, B. Ko, A. Beygelzimer, and V. Madduri. Topology Discovery and Link State Detection using Routing Events. under submission.

[26] Tao Wan, Evangelos Kranakis, and P.C. van Oorschot. Pretty Secure BGP (psBGP). In *The 12th Annual Network and Distributed System Security Symposium*, February 2005.

[27] Russ White. Securing BGP Through Secure Origin BGP. *Internet Protocol Journal*, 6(3), September 2003.

[28] M. Zafer, B. Ko, and I. W. Ho. Cooperative Transmit-Power Estimation under Wireless Fading. In *ACM Mobihoc 2008*, Hong Kong, May 2008.